INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS

Int. J. Numer. Meth. Fluids 2010; **00**:1–**??**

Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld

Dynamically Correct Formulations of the Linearised Navier-Stokes Equations

O. J. Dellar* and B. Ll. Jones

Department of Automatic Control and Systems Engineering, The University of Sheffield, S1 3JD, UK

SUMMARY

Motivated by the need to efficiently obtain low-order models of fluid flows around complex geometries for the purpose of feedback control system design, this paper considers the effect on system dynamics of basing plant models on different formulations of the linearised Navier-Stokes equations. We consider the dynamics of a single computational node formed by spatial discretisation of the governing equations in both primitive variables (momentum equation & continuity equation) and pressure Poisson equation (PPE) formulations. This reveals fundamental numerical differences at the nodal level, whose effects on the system dynamics at the full system level are exemplified by considering the corresponding formulations of a two-dimensional (2D) channel flow, subjected to a variety of different boundary conditions.

Received ...

KEY WORDS: Navier-Stokes; Finite difference; Incompressible flow; Partial differential equations; Reduced order modelling; spectral

1. INTRODUCTION

This work is motivated by the need to design feedback control systems to reduce pressure drag acting on road vehicles, which within the UK accounted for 28% of total annual energy consumption in 2014, and whose CO_2 emissions rose by 26% between 1990 and 2004 [1]. At motorway speeds, aerodynamic drag arising from vortex shedding over the bluff rear end of the vehicle is responsible for up to two thirds of this energy consumption [2, 3]. Initial numerical studies of prototype bluff body flows have shown the potential for feedback control to suppress vortex shedding, thus leading to an overall increase in base mean pressure. This can be achieved using, for example, zero-net-mass-flux (ZNMF) slot jet actuation on the body's trailing edge in response to pressure measurements on the rear face [4]. Whilst this can also be achieved with open-loop control, such as fixed frequency/amplitude periodic forcing, feedback control has the important advantages of

*Correspondence to: O. J. Dellar, Department of Automatic Control and Systems Engineering, The University of Sheffield, S1 3JD, UK. E-mail: ojdellar1@sheffield.ac.uk.

allowing a system to be robust to the effects of uncertainty [5], such as may arise from unmodelled external disturbances, system dynamics and changing operating conditions.

The problem faced with designing conventional feedback controllers is that they require linear plant models that are of low-order, in the sense that the state dimension is $\mathcal{O} \lesssim 10^2$. Obtaining such models from the underlying Navier-Stokes equations is non-trivial and requires approximation, given that the latter are nonlinear and infinite-dimensional. If the objective of the control system is to suppress perturbations around a mean flow, then designing feedback controllers from linearised approximations can give rise to acceptable closed-loop performance, as has been shown in a number of studies (see, e.g. [6, 7, 8, 9, 10, 4, 11, 12, 13]). However, obtaining low-order approximations that are accurate in the sense of retaining the important dynamics of a given flow remains an open problem [12]. A typical approach relies upon spatial discretisation of the linearised Navier-Stokes equations on a computational mesh in order to obtain finite-dimensional state-space models [14, 15, 16, 13].

The construction of state-space models of a flow by spatial discretisation is typically restricted to simple geometries such as plane channel flow, whereby the assumption of periodicity in the streamwise (and spanwise for three-dimensional geometries) direction allows the use of Fourier transforms in space, decoupling the flow by spatial wavenumber [6]. For more complex geometries such assumptions are less readily applicable, and hence computational fluid dynamics (CFD) codes often use methods based on local interpolants, such as finite difference, finite volume, or finite element discretisation. Whilst these can be used for constructing the state-space matrices describing complex geometry flows (such as over a bluff-body vehicle), it is difficult in practice, and typically results in systems with extremely large ($\mathcal{O} > 10^5$) state dimension, giving rise to impractically large system matrices. However, the vast majority of these states have little influence over the actuator to sensor response of the flow system, which means that satisfactory models for feedback control design can be many orders of magnitude smaller, in terms of state dimension. In the context of bluff-body drag reduction, Dahan *et al*. [4] showed the underlying input-output frequency response of some flows actually resemble those of *second*-order systems.

An approach which avoids constructing large-scale system matrices and lends itself to exposing the potentially low-order nature of a fluid flow was proposed in [17, 15] and was applied to 2D channel flow. Finite differences and a Chebyshev expansion were used to discretise in the streamwise and wall-normal directions, respectively. This yielded state-space subsystem models at each streamwise location which were then used to compute the local frequency response, and finally the Redheffer star product [18] was used to chain the local frequency responses together between neighbouring subsystems, producing the overall global system frequency response. With this, low-order transfer function models can be fitted to the data, and used for control design with guaranteed closed-loop stability margins [19].

Whilst this approach lends itself to the modelling of complex geometry flows, the choice of formulation and discretisation of the linearised Navier-Stokes equations is important. Centred finite differences are attractive since this results in simple connections between neighbouring subsystems, however it is well known that directly discretising the Navier-Stokes equations in their standard momentum/continuity formulation on a co-located mesh using finite differences can lead to the so-called ‘checkerboard instability’, whereby non-physical sawtooth shaped pressure fields can satisfy the discretised equations [20, 21, 22, 23]. Figure 1 shows an example of such a pressure field.

DOI: 10.1002/fld

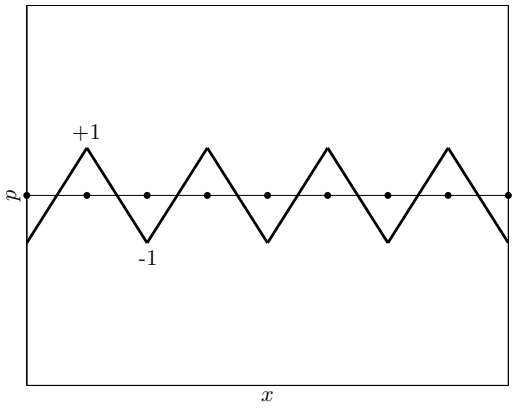

Figure 1. Example of checkerboard instability; non-physical pressure field satisfying discretised incompressible Navier-Stokes equations (discretised using centred finite differences), since $\partial p/\partial x = 0$ at all computational nodes (black dots).

The checkerboard instability issue is a result of the fact that after any spatial discretisation the incompressible Navier-Stokes equations constitute a system of differential algebraic equations (DAEs) of 'higher index' [24], with only the pressure gradient appearing in the equations, and not pressure itself. The nonlinear system is of differentiation index two [25], or strangeness index one [24, 26], since at least part of the equations must be differentiated a minimum of two times with respect to the time variable in order to obtain a system of ordinary differential equations (ODEs). One way of circumventing the checkerboard instability issue is by reformulating the set of equations as a strangeness free system [24]. This can be achieved by reformulating the system in, for example, velocity and vorticity variables, which results in a system of ODEs (of differentiation index zero [24]), or supplementing the original equations with a pressure Poisson equation (PPE) which results in a strangeness free DAE system (of differentiation index one).

Whilst supplementing the original system of equations with the PPE avoids the checkerboard instability, it poses the additional problem of choosing a correct pressure boundary condition. For the case of the nonlinear incompressible Navier-Stokes equations, there seems to be some disagreement in the literature regarding this matter [27, 28, 29, 30, 31, 32], and so in this paper we consider several different boundary conditions, and study their effects on linearised system dynamics.

Another way of circumventing the issue of checkerboard instability, which avoids the need to reformulate the governing equations as a strangeness free system, is by discretising the original formulation of the equations on a staggered mesh, such as the 'marker and cell' method [33]. This method does, however, come with its own difficulties. Namely that one typically obtains discretisations where pressure nodes do not lie on the domain boundaries, which is inconvenient from a control design perspective if pressure on a surface is the measured signal, as is typically the case for bluff-body drag reduction.

For the purpose of simulation, the various formulations of the Navier-Stokes equations have received much attention in the literature, however for the purpose of feedback control design this is not the case. In particular, for simulation purposes the equations can be formulated in such a way that a 'pseudopressure' which requires only simple boundary conditions is used rather than

DOI: 10.1002/fld

actual thermodynamic pressure, for example in fractional step/projection methods [34, 30]. This allows efficient simulation by first computing an intermediate velocity field at each timestep which does not, in general, satisfy the divergence-free requirement of the governing equations, before correcting this velocity field using the pseudopressure, yielding a divergence-free field [30]. In the context of simulation, the validity of numerical schemes is typically assessed by comparing certain time averaged properties, such as mean velocity profiles, to benchmark data from other simulations or experiments (see, e.g. [35, 36, 37, 4, 38]). On the other hand, models derived for control design must correctly capture the continuous time dynamics of both velocity and pressure fields in a single set of differential algebraic equations, and in particular must exhibit the correct frequency response, particularly around the unity gain crossover frequency.

This paper reconciles current understanding of different formulations of the linearised Navier-Stokes equations using frequency domain analysis in order to compare the system dynamics obtained from different model formulations. This essentially provides understanding of how to correctly formulate and discretise the linearised Navier-Stokes equations for the purpose of designing feedback controllers. In Section 2 the governing equations in momentum/continuity and PPE formulations are presented, before considering transfer function representations of individual computational node subsystems in Section 3. In Section 4 different formulations of the linearised incompressible Navier-Stokes equations are considered, and PPE formulations with pressure boundary conditions based on several of those suggested in the literature are used in order to construct state-space models of a 2D channel flow. The different models' dynamic similarities/differences are analysed by considering pole-zero plots, system eigenfunctions, energy-weighted pseudospectra, frequency response, and the ν-gap metric between each model formulation and a benchmark model. Section 5 provides concluding remarks.

2. GOVERNING EQUATIONS FOR INCOMPRESSIBLE FLOW

The remainder of this paper considers the linearised incompressible Navier-Stokes equations.

$$\frac{\partial \mathbf{u}'(\mathbf{x},t)}{\partial t} + \mathbf{u}'(\mathbf{x},t)\cdot\nabla\bar{\mathbf{u}}(\mathbf{x}) + \bar{\mathbf{u}}(\mathbf{x})\cdot\nabla\mathbf{u}'(\mathbf{x},t) = -\nabla p'(\mathbf{x},t) + \frac{1}{\mathrm{Re}}\nabla^2\mathbf{u}'(\mathbf{x},t), \tag{1a}$$

$$\nabla\cdot\mathbf{u}'(\mathbf{x},t) = 0, \tag{1b}$$

where velocity and pressure have been decomposed into a temporally averaged mean part and a time dependent fluctuating part, e.g. $\mathbf{u}(\mathbf{x},t) := \bar{\mathbf{u}}(\mathbf{x}) + \mathbf{u}'(\mathbf{x},t)$, $\bar{\mathbf{u}}(\cdot) : \Omega \to \mathbb{R}^2$ is the mean velocity field with components $(\bar{u},\bar{v})$, $\mathbf{u}'(\cdot,\cdot) : \Omega\times[0,t_f] \to \mathbb{R}^2$ is the fluctuating velocity field with components (u',v'), $p'(\cdot,\cdot) : \Omega\times[0,t_f] \to \mathbb{R}$ is the fluctuating pressure field, $\Omega \subset \mathbb{R}^2$ is the spatial domain with boundary $\partial\Omega$, $t_f \in \mathbb{R}^+$ is the end of the time interval, and $\mathbf{x} := (x,y) \in \Omega$ is a point in the domain. The Reynolds number of the flow is $\mathrm{Re} \in \mathbb{R}$. In the interests of simplicity, this paper considers 2D flows, but the theory readily extends to the 3D case.

In this work we consider two formulations of the linearised Navier-Stokes equations. The first of these is the primitive variables formulation, given by the momentum (1a) and continuity (1b) equations [30]. The presence of the algebraic constraint in the form of the continuity equation makes this a set of partial differential algebraic equations (PDAEs). As was discussed in Section 1,

Prepared using fldauth.cls
DOI: 10.1002/fld

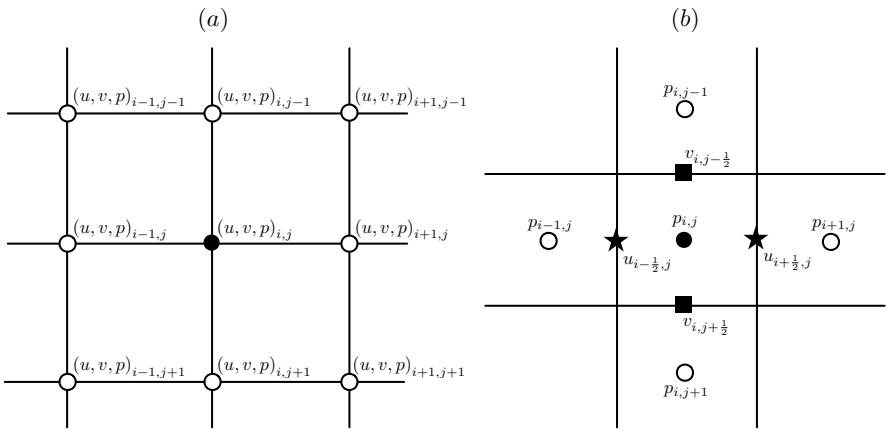

Figure 2. Computational meshes: (a) co-located computational mesh; (b) staggered mesh.

this PDAE system of differentiation index two can be reformulated as a strangeness-free system by introducing the PPE. The linearised PPE is obtained by substituting (1b) into the divergence of (1a) [21], yielding

$$\nabla^2 p'(\mathbf{x},t) = -\nabla\cdot(\mathbf{u}'(\mathbf{x},t)\cdot\nabla\bar{\mathbf{u}}(\mathbf{x})) - \nabla\cdot(\bar{\mathbf{u}}(\mathbf{x})\cdot\nabla\mathbf{u}'(\mathbf{x},t))\,,\ \forall\,(\mathbf{x},t)\in\Omega\times[0,t_f]\,. \tag{1c}$$

The system of equations comprising (1a) and (1c) is known as the PPE formulation [30] and represents the second formulation employed in this work.

Boundary and initial conditions for the primitive variables formulation are given by

$$\bar{\mathbf{u}}(\mathbf{x}) = \bar{\mathbf{u}}_{\partial\Omega}(\mathbf{x}),\ \forall\,\mathbf{x}\in\partial\Omega, \tag{2a}$$

$$\mathbf{u}'(\mathbf{x},t) = \mathbf{u}'_{\partial\Omega}(\mathbf{x},t),\ \forall\,(\mathbf{x},t)\in\partial\Omega\times[0,t_f]\,, \tag{2b}$$

$$\mathbf{u}'(\mathbf{x},0) = \mathbf{u}'_0(\mathbf{x}),\ \forall\,\mathbf{x}\in\Omega. \tag{2c}$$

We defer discussion of the boundary conditions for the PPE formulation until Section 3.2.

3. INDIVIDUAL COMPUTATIONAL NODE SUBSYSTEMS

In the following, the primitive variables formulation is discretised on both a co-located mesh and a staggered mesh, the PPE formulation is discretised on a co-located mesh only, and state-space models describing the dynamics of a single computational node are constructed. The co-located mesh, in which both velocity and pressure values are stored on the cell vertices, is depicted in Figure 2(a). The staggered mesh is shown in Figure 2(b), in which the velocity components u and v are stored on the vertical and horizontal cell boundaries, respectively, whilst the pressure is stored in the cell centre.

DOI: 10.1002/fld

The descriptor state-space models considered in this work are of the form

$$E\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = A\boldsymbol{\chi}(t) + B\boldsymbol{\xi}(t), \tag{3a}$$

$$\mathbf{y}(t) = C\boldsymbol{\chi}(t), \tag{3b}$$

where $\boldsymbol{\chi}(t) \in \mathbb{R}^n$ is the state vector, $\boldsymbol{\xi}(t) \in \mathbb{R}^q$ is the vector of system inputs, and $\mathbf{y}(t) \in \mathbb{R}^p$ is the vector of system outputs. $E, A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times q}$, and $C \in \mathbb{R}^{p\times n}$ are matrices describing the dynamics of the system, how inputs affect the states, and how outputs are defined as linear combinations of the states, respectively. State-space models arising from spatial discretisation of PDAE systems typically yield rank deficient E matrices due to algebraic constraints, arising from (1b), (1c), and associated boundary conditions.

Taking the Laplace transform of (3a) and inserting it into the Laplace transform of (3b) yields the expression

$$\tilde{\mathbf{y}}(s) = \underbrace{C\left(sE - A\right)^{-1}B}_{G(s)}\tilde{\boldsymbol{\xi}}(s), \tag{4}$$

where $G(s) \in \mathscr{R}^{p\times q}$ is the real-rational transfer function matrix from inputs to outputs, and $s \in \mathbb{C}$. The Laplace transforms of $\mathbf{y}(t)$ and $\boldsymbol{\xi}(t)$ are $\tilde{\mathbf{y}}(s) \in \mathbb{C}^p$ and $\tilde{\boldsymbol{\xi}}(s) \in \mathbb{C}^q$, respectively.

The poles of a linear dynamical system determine its asymptotic behaviour and are defined as

$$\mathcal{P}\left(G(s)\right) := \left\{s \in \mathbb{C} : \det\left(sE - A\right) = 0\right\}. \tag{5}$$

The zeros are the values of $s \in \mathbb{C}$ at which the system output drops to zero despite the input and states being non-zero [39, 40]. The set of zeros is thus defined as

$$\mathcal{Z}\left(G(s)\right) := \left\{s \in \mathbb{C} : \mathbf{y}(t) = 0,\ \boldsymbol{\xi}(t), \boldsymbol{\chi}(t) \neq 0\right\}. \tag{6}$$

The presence and locations of the zeros are dependent on sensor and actuation arrangement, and can impose fundamental limitations on the performance achievable through feedback control [40]. For single-input single-output systems, the poles and zeros are simply equal to the roots of the denominator and numerator of the transfer function, respectively.

The different formulations of the governing equations are discretised in space on computational meshes with grid spacing δ in both the x and y directions. Second-order accurate centred finite difference schemes are used for spatial derivatives as this yields simple interconnections between neighbouring node subsystems, which enables efficient evaluation of system frequency response by chaining nodes together, as demonstrated in [17, 15]. Such finite difference schemes are of the form

$$\frac{\partial f_{i,j}}{\partial x} = \frac{f_{i+1,j} - f_{i-1,j}}{2\delta} + \mathcal{O}\left(\delta^2\right),$$

$$\frac{\partial^2 f_{i,j}}{\partial x^2} = \frac{f_{i-1,j} - 2f_{i,j} + f_{i+1,j}}{\delta^2} + \mathcal{O}\left(\delta^2\right),$$

where i and j denote the computational node's indices in the x and y directions, respectively. For the case of co-located mesh discretisations, $u'_{i,j}(t)$, $v'_{i,j}(t)$, and $p'_{i,j}(t)$ are grouped as

DOI: 10.1002/fld

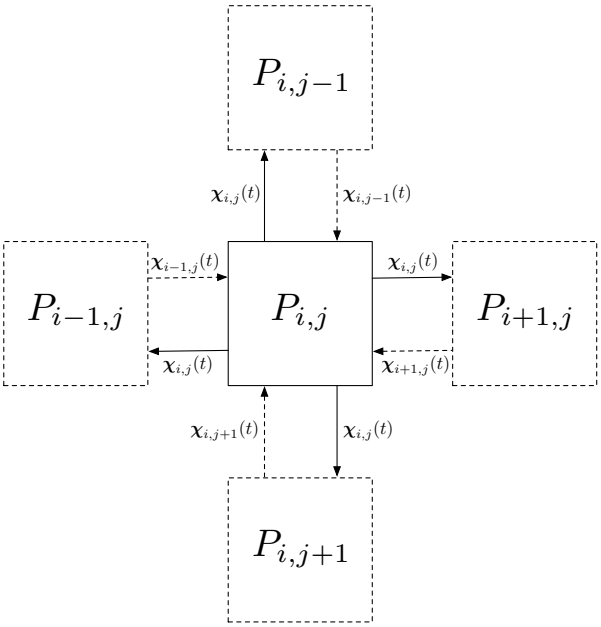

Figure 3. Connections between computational node subsystems. The $(i,j)^{\text{th}}$ subsystem is denoted $P_{i,j}$.

a single subsystem, and will be referred to as the $(i,j)^{\text{th}}$ node, whilst for staggered mesh discretisations, $u'_{i+1/2,j}(t)$, $v'_{i,j+1/2}(t)$, and $p'_{i,j}(t)$ are grouped together as a single subsystem, and will also be referred to as the $(i,j)^{\text{th}}$ node (where the indices of the pressure node are used to label the subsystem).

The dynamics of the $(i,j)^{\text{th}}$ computational node subsystem can be expressed as a descriptor state-space system whose inputs are the state values from its neighbouring nodes, and outputs are the $(i,j)^{\text{th}}$ node subsystem's state, as depicted in Figure 3. The corresponding state and output equations are

$$E_{i,j}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}_{i,j}(t) = A_{i,j}\boldsymbol{\chi}_{i,j}(t) + B_{i,j}\boldsymbol{\xi}_{i,j}(t), \tag{7a}$$

$$\mathbf{y}_{i,j}(t) = C_{i,j}\boldsymbol{\chi}_{i,j}(t), \tag{7b}$$

where $\boldsymbol{\chi}_{i,j} := \begin{bmatrix} u'_{i,j} & v'_{i,j} & p'_{i,j} \end{bmatrix}^\top \in \mathbb{R}^3$ is the subsystem's state vector in the co-located case, and $\boldsymbol{\chi}_{i,j} := \begin{bmatrix} u'_{i+1/2,j} & v'_{i,j+1/2} & p'_{i,j} \end{bmatrix}^\top \in \mathbb{R}^3$ is the subsystem's state vector in the staggered case. The vector of state values flowing in from the neighbouring subsystems is $\boldsymbol{\xi}_{i,j} := \begin{bmatrix} \boldsymbol{\chi}_{i,j-1}^\top & \boldsymbol{\chi}_{i+1,j}^\top & \boldsymbol{\chi}_{i,j+1}^\top & \boldsymbol{\chi}_{i-1,j}^\top \end{bmatrix}^\top \in \mathbb{R}^{12}$, and $\mathbf{y}_{i,j} \in \mathbb{R}^3$ is the subsystem's output. The matrices $E_{i,j}, A_{i,j} \in \mathbb{R}^{3\times 3}$, $B_{i,j} \in \mathbb{R}^{3\times 12}$ and $C_{i,j} \in \mathbb{R}^{3\times 3}$ arise from the spatial discretisation of the governing equations, and have different structure depending on the particular formulation/discretisation.

DOI: 10.1002/fld

3.1. Primitive Variables Formulation

In the following, the structure of the state-space matrices describing a single computational node subsystem are presented for three different cases.

3.1.1. Case I: Upon discretising the standard primitive variables formulation using finite differences on a co-located mesh, the $E_{i,j}$ and $A_{i,j}$ matrices are found to have the following structure:

$$E_{i,j} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; A_{i,j} := \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{8}$$

The elements of $A_{i,j}$ are given in Appendix A.1. When attempting to compute the node's transfer function as in (4), it is immediately clear that $(sE_{i,j} - A_{i,j})$ is singular (rank deficient) for all $s \in \mathbb{C}$, and so the resolvent $(sE_{i,j} - A_{i,j})^{-1}$ and transfer function $G_{i,j}(s)$, do not exist. This lack of uniqueness is the source of the checkerboard instability.

3.1.2. Case II: Discretising the same primitive variables formulation on a staggered mesh yields $E_{i,j}$ and $A_{i,j}$ matrices with the following structure:

$$E_{i,j} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; A_{i,j} := \begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & a_{23} \\ a_{31} & a_{32} & 0 \end{bmatrix}. \tag{9}$$

Appendix A.2 gives the elements of $A_{i,j}$. In this case, the matrix pair $(E_{i,j}, A_{i,j})$ is regular (in that there exists $s \in \mathbb{C}$ such that $(sE_{i,j} - A_{i,j})$ is non-singular [41]), and so the resolvent $(sE_{i,j} - A_{i,j})^{-1}$, and hence transfer function $G_{i,j}(s)$, exist and are uniquely defined.

3.1.3. Case III: Finally, discretising the PPE formulation on a co-located mesh yields $E_{i,j}$ and $A_{i,j}$ matrices with the following structure:

$$E_{i,j} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; A_{i,j} := \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix}. \tag{10}$$

The elements of $A_{i,j}$ are given in Appendix A.3. For the flows considered in this paper, this again yields a regular matrix pair $(E_{i,j}, A_{i,j})$, resulting in the existence and uniqueness of both the resolvent $(sE_{i,j} - A_{i,j})^{-1}$, and transfer function $G_{i,j}(s)$.

The existence of subsystem transfer functions for the cases of the primitive variables formulation discretised on a staggered mesh and the PPE formulation discretised on a co-located mesh, shows that these formulations are well posed, whilst the contrary applies for the case of the primitive variables formulation discretised on a co-located mesh.

DOI: 10.1002/fld

3.2. PPE Pressure Boundary Conditions

Although the PPE formulation discretised on a co-located mesh is well posed, the main difficulty with such a formulation lies in the correct choice of pressure boundary conditions. Several different pressure boundary conditions present in the literature are now considered.

The first, and simplest pressure boundary condition considered is the Neumann type condition

$$\frac{\partial p'(\mathbf{x},t)}{\partial \mathbf{n}} = 0, \ \forall \ (\mathbf{x},t) \in \partial\Omega \times [0,t_f]\,, \tag{11}$$

where $\mathbf{n} \in \mathbb{R}^2$ is the normal outward pointing unit vector. As stated in [30], the wall-normal gradient of the pressure at the boundary in incompressible flow is, in general, non-zero, however if treated with care it can be used as part of fractional step methods [34] for the purposes of simulation.

For the case of the incompressible nonlinear Navier-Stokes equations, it was suggested by Gresho and Sani [27] that the correct pressure boundary condition for a PPE formulation is a Neumann type condition obtained by projecting the momentum equation onto the boundary in the normal direction. Applying this idea to the linearised equations yields the boundary condition

$$\begin{aligned}\mathbf{n} \cdot \nabla p'(\mathbf{x},t) = \mathbf{n} \cdot \Bigg(& \frac{1}{\mathrm{Re}} \nabla^2 \mathbf{u}'(\mathbf{x},t) - \mathbf{u}'(\mathbf{x},t) \cdot \nabla \bar{\mathbf{u}}(\mathbf{x},t) \\ & - \bar{\mathbf{u}}(\mathbf{x},t) \cdot \nabla \mathbf{u}'(\mathbf{x},t) - \frac{\partial}{\partial t} \mathbf{u}'(\mathbf{x},t) \Bigg), \ \forall \ (\mathbf{x},t) \in \partial\Omega \times [0,t_f]\,.\end{aligned} \tag{12}$$

For the common case of a no-slip boundary this condition reduces to

$$\mathbf{n} \cdot \nabla p'(\mathbf{x},t) = \mathbf{n} \cdot \frac{1}{\mathrm{Re}} \nabla^2 \mathbf{u}'(\mathbf{x},t), \ \forall \ (\mathbf{x},t) \in \partial\Omega \times [0,t_f]\,. \tag{13}$$

Whilst this seems a more logical approach to deducing a valid pressure boundary condition than that discussed above, Rempfer [42, 30] points out that using the PPE in conjunction with this condition yields a system of equations where the divergence free condition (1b) appears nowhere. For the nonlinear case, Rempfer [30] shows that substituting the PPE into the divergence of the momentum equation yields a heat equation. For the case of the linearised equations, this amounts to substituting (1c) into the divergence of (1a), yielding

$$\frac{\partial}{\partial t} \Theta(\mathbf{x},t) = \frac{1}{\mathrm{Re}} \nabla^2 \Theta(\mathbf{x},t), \ \forall \ (\mathbf{x},t) \in \Omega \times [0,t_f]\,, \tag{14}$$

where $\Theta(\mathbf{x},t) := \nabla \cdot \mathbf{u}'(\mathbf{x},t)$ is the divergence of the velocity field. This shows that the combination of (1a) and (1c) does not enforce the incompressibility of the velocity field, it just specifies that the divergence of the velocity field satisfies a heat equation, and is a harmonic funtion in the steady-state case. According to the extremum theorems for harmonic functions [43], harmonic functions assume their extremal values on the boundary $\partial\Omega$, which led to Rempfer's theorem regarding the nonlinear case [30, Thm.1], stating that the solution to the incompressible Navier-Stokes equations in primitive variables formulation is equivalent to the solution of the PPE formulation if and only if the boundary conditions for the PPE are chosen such that $\nabla \cdot \mathbf{u}(\mathbf{x},t) = 0, \ \forall \ (\mathbf{x},t) \in \partial\Omega \times [0,t_f]$. This was described by Rempfer [42, 30] as an 'indirect boundary condition', since the boundary

DOI: 10.1002/fld

condition for one variable, in this case pressure, must be chosen such that some condition for another variable, in this case velocity, is met on the boundary.

In more recent work, Shirokoff and Rosales [31] took a new approach to choosing not only the PPE pressure boundary condition, but also the corresponding velocity boundary conditions for a no-slip boundary. Rather than enforcing all d components of the no-slip boundary condition on the nonlinear momentum equation (where $d \in \{2, 3\}$ is the number of spatial dimensions), only the $d-1$ tangential components are enforced, and the set of boundary conditions for the momentum equation is completed with the divergence free condition (1b). In practice, this results in specifying that the $d-1$ tangential velocity components are equal to zero on the boundary, and that the normal gradient of the normal velocity component is also equal to zero, hence satisfying (1b). For the linearised equations this becomes

$$\boldsymbol{\tau} \cdot \mathbf{u}'(\mathbf{x}, t) = 0, \ \forall \ (\mathbf{x}, t) \in \partial\Omega \times [0, t_f] \,, \tag{15a}$$

$$\mathbf{n} \cdot (\nabla (\mathbf{n} \cdot \mathbf{u}'(\mathbf{x}, t))) = 0, \ \forall \ (\mathbf{x}, t) \in \partial\Omega \times [0, t_f] \,, \tag{15b}$$

where $\boldsymbol{\tau} \in \mathbb{R}^2$ is the wall-tangential unit vector. As for the pressure boundary condition, the momentum equation is projected in the normal direction onto the boundary yielding the Neumann type condition (13).

4. A NUMERICAL INVESTIGATION OF THE DYNAMICS OF 2D CHANNEL FLOW

Analysis of the structure of the system matrices describing a single computational node subsystem revealed, from a systems theory perspective, the unsuitability of the standard primitive variables formulation on a co-located mesh discretised using finite differences. However, such analysis revealed little concerning the suitability for feedback control system design, based on models obtained from different formulations employing various boundary conditions. In order to gain insight into this, the dynamics of a full fluid flow system must be considered.

In the following, the benchmark example of a fully developed flow between two flat plates is considered. This is one of very few fluid flows for which an analytical solution to the Navier-Stokes equations is known, and for which the flow dynamics have been studied extensively (see, e.g. [44, 45, 46, 47]).

After non-dimensionalising length scales by the channel half-height h, velocity scales by the maximum laminar centreline velocity $\bar{u}_0$, and pressure by $\rho \bar{u}_0^2$, where ρ is the density of the fluid, the upper walls of the channel are located at $y = \pm 1$, and the streamwise (x) channel length is $l = 2\pi$, as depicted in Figure 4. It can be shown that the the baseline velocity is independent of x, and is equal to $\bar{u}(y) = 1 - y^2$. Hence, the linearised Navier-Stokes equations in primitive variables formulation

DOI: 10.1002/fld

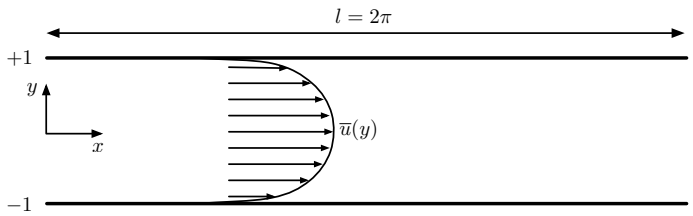

Figure 4. Schematic of 2D channel flow geometry.

reduce to

$$\frac{\partial u'(\mathbf{x},t)}{\partial t} + \left(1-y^2\right)\frac{\partial u'(\mathbf{x},t)}{\partial x} - 2yv'(\mathbf{x},t) = -\frac{\partial p'(\mathbf{x},t)}{\partial x} + \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 u'(\mathbf{x},t)}{\partial x^2} + \frac{\partial^2 u'(\mathbf{x},t)}{\partial y^2}\right), \tag{16a}$$

$$\frac{\partial v'(\mathbf{x},t)}{\partial t} + \left(1-y^2\right)\frac{\partial v'(\mathbf{x},t)}{\partial x} = -\frac{\partial p'(\mathbf{x},t)}{\partial y} + \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 v'(\mathbf{x},t)}{\partial x^2} + \frac{\partial^2 v'(\mathbf{x},t)}{\partial y^2}\right), \tag{16b}$$

$$\frac{\partial u'(\mathbf{x},t)}{\partial x} + \frac{\partial v'(\mathbf{x},t)}{\partial y} = 0, \tag{16c}$$

where the Reynolds number is defined as $\mathrm{Re} := \bar{u}_0 h/\nu$.

The PPE (1c) reduces to

$$\frac{\partial^2 p'(\mathbf{x},t)}{\partial x^2} + \frac{\partial^2 p'(\mathbf{x},t)}{\partial y^2} = 4y\frac{\partial v'(\mathbf{x},t)}{\partial x}. \tag{17}$$

No-slip boundary conditions are assumed on the upper and lower walls, $u'(x,\pm 1,t) = v'(x,\pm 1,t) = 0$, and boundary conditions in the streamwise direction are assumed periodic, $u'(0,y,t) = u'(l,y,t)$, $v'(0,y,t) = v'(l,y,t)$, and $p'(0,y,t) = p'(l,y,t)$.

With these equations, finite-dimensional state-space models will be constructed which describe the dynamics of the perturbation variables about the baseline flow.

4.1. Orr-Sommerfeld State-space Model

Firstly, the well known Orr-Sommerfeld formulation [46, 6, 48] will be used to obtain a benchmark state-space model of the flow, against which other formulations can be compared.

Substituting (16c) along with the divergence of (16a) and (16b) into the Laplacian of (16b) yields the following 4$^{\text{th}}$ order PDE, where the pressure perturbation variable $p'(\mathbf{x},t)$, has been analytically removed:

$$\nabla^2\frac{\partial}{\partial t}v'(\mathbf{x},t) + \left(1-y^2\right)\frac{\partial}{\partial x}\nabla^2 v'(\mathbf{x},t) + 2\frac{\partial}{\partial x}v'(\mathbf{x},t) = \frac{1}{\mathrm{Re}}\nabla^4 v'(\mathbf{x},t). \tag{18}$$

Exploiting the periodicity in the streamwise direction, Fourier transforms are taken in x. This allows the y direction perturbation velocity to be written

$$v'(\mathbf{x},t) := \hat{v}(y,t)e^{\mathrm{i}\alpha x}, \tag{19}$$

DOI: 10.1002/fld

where $\mathrm{i} := \sqrt{-1}$, and $\hat{v}(y,t) \in \mathbb{C}$ is the Fourier transformed wall-normal velocity perturbation at an individual spatial wavenumber $\alpha \in \mathbb{R}$. Spatial derivatives with respect to x in (18) simply become multiplication by $\mathrm{i}\alpha$, and (18) can be written

$$\mathcal{E}\frac{\partial}{\partial t}\hat{v}(y,t) = \mathcal{A}\hat{v}(y,t), \tag{20}$$

where the system operators are defined as

$$\mathcal{E} := \mathrm{Re}\left(\frac{\partial^2}{\partial y^2} - \alpha^2\right), \tag{21a}$$

$$\mathcal{A} := \frac{\partial^4}{\partial y^4} - 2\alpha^2\frac{\partial^2}{\partial y^2} + \alpha^4 - \mathrm{i}\alpha\mathrm{Re}\left(\left(1-y^2\right)\left(\frac{\partial^2}{\partial y^2} - \alpha^2\right) + 2\right). \tag{21b}$$

As is standard for channel flow geometries, discretisation in the wall-normal direction is performed using Chebyshev interpolants. As such, (20) is discretised upon $n_y \in \mathbb{N}$ Chebyshev collocation points and the wall-normal differential operators are approximated by differentiation matrices [49], yielding the following finite-dimensional descriptor state equation

$$E\frac{\mathrm{d}}{\mathrm{d}t}\vec{\hat{v}}(t) = A\vec{\hat{v}}(t), \tag{22}$$

where $E, A \in \mathbb{C}^{n_y \times n_y}$ are discrete versions of $\mathcal{E}$ and $\mathcal{A}$, respectively, and $\vec{\hat{v}}(t) \in \mathbb{C}^{n_y}$ is the state vector comprising the values of $\hat{v}(y,t)$ evaluated on the collocation points. Finally, premultiplying (22) by E^{-1} yields the following state equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\vec{\hat{v}}(t) = A_{\mathrm{OS}}\vec{\hat{v}}(t), \tag{23}$$

where $A_{\mathrm{OS}} := E^{-1}A$ is the system dynamics matrix.

The poles of the Orr-Sommerfeld system correspond to the eigenvalues of A_{OS}, and are dependent on the Reynolds number and streamwise wavenumber. For the case $(\mathrm{Re}, \alpha) = (5772.22, 1.02)$, the poles of the system are plotted in Figure 5(a) and are in agreement with those originally calculated in [50]. For clarity of exposition, the three main branches of the spectrum are labelled A, S and P, as in [51]. The pole with greatest real part (corresponding to the least stable mode) is shown in green, along with its mode shape (eigenfunction) in Figure 5(b).

4.2. Primitive Variables and PPE Formulation State-space Models

State-space models of 2D channel flow based on both primitive variables and PPE formulations are now constructed. Starting with (16) and (17), once again the periodicity in the streamwise direction is exploited and Fourier transforms are taken in x, such that the variables can be written

$$u'(\mathbf{x},t) := \hat{u}(y,t)e^{\mathrm{i}\alpha x}, \quad v'(\mathbf{x},t) := \hat{v}(y,t)e^{\mathrm{i}\alpha x}, \quad p'(\mathbf{x},t) := \hat{p}(y,t)e^{\mathrm{i}\alpha x}. \tag{24}$$

DOI: 10.1002/fld

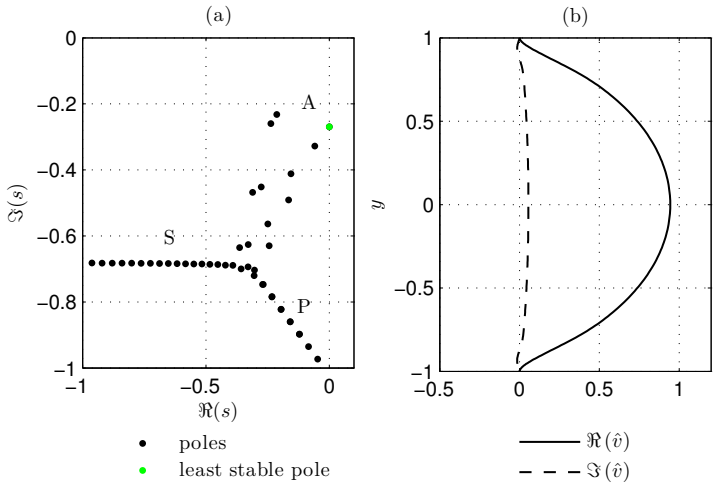

Figure 5. Poles (a) and eigenfunction corresponding to least stable mode (b) of Orr-Sommerfeld system.

Evaluating the derivatives with respect to x in (16) and (17) yields two systems of PDAEs. The first of these is the following primitive variables formulation:

$$\underbrace{\begin{bmatrix}\mathcal{I} & 0 & 0\\ 0 & \mathcal{I} & 0\\ 0 & 0 & 0\end{bmatrix}}_{\mathcal{E}_{\mathrm{Pr}} \in \mathbb{R}^{3\times 3}} \frac{\partial}{\partial t} \underbrace{\begin{bmatrix}\hat{u}(y,t)\\ \hat{v}(y,t)\\ \hat{p}(y,t)\end{bmatrix}}_{\mathbf{X}(y,t) \in \mathbb{C}^3} = \underbrace{\begin{bmatrix}\mathcal{A}_{11} & \mathcal{A}_{12} & \mathcal{A}_{13}\\ 0 & \mathcal{A}_{22} & \mathcal{A}_{23}\\ \mathcal{A}_{31} & \mathcal{A}_{32} & 0\end{bmatrix}}_{\mathcal{A}_{\mathrm{Pr}} \in \mathbb{C}^{3\times 3}} \begin{bmatrix}\hat{u}(y,t)\\ \hat{v}(y,t)\\ \hat{p}(y,t)\end{bmatrix}, \tag{25}$$

where the system operators in $\mathcal{A}_{\mathrm{Pr}}$ are defined as

$$\mathcal{A}_{11} = A_{22} := -\mathrm{i}\alpha\left(1-y^2\right) + \frac{1}{\mathrm{Re}}\left(\frac{\partial^2}{\partial y^2} - \alpha^2\right), \tag{26a}$$

$$\mathcal{A}_{12} := 2y, \tag{26b}$$

$$\mathcal{A}_{13} := -\mathrm{i}\alpha, \tag{26c}$$

$$\mathcal{A}_{23} := -\frac{\partial}{\partial y}, \tag{26d}$$

$$\mathcal{A}_{31} := \mathrm{i}\alpha, \tag{26e}$$

$$\mathcal{A}_{32} := \frac{\partial}{\partial y}, \tag{26f}$$

and $\mathcal{I}$ is the identity operator.

The second system of PDAEs arises from the following PPE formulation:

$$\underbrace{\begin{bmatrix}\mathcal{I} & 0 & 0\\ 0 & \mathcal{I} & 0\\ 0 & 0 & 0\end{bmatrix}}_{\mathcal{E}_{\mathrm{PPE}} \in \mathbb{R}^{3\times 3}} \frac{\partial}{\partial t} \underbrace{\begin{bmatrix}\hat{u}(y,t)\\ \hat{v}(y,t)\\ \hat{p}(y,t)\end{bmatrix}}_{\mathbf{X}(y,t) \in \mathbb{C}^3} = \underbrace{\begin{bmatrix}\mathcal{A}_{11} & \mathcal{A}_{12} & \mathcal{A}_{13}\\ 0 & \mathcal{A}_{22} & \mathcal{A}_{23}\\ 0 & \mathcal{A}_{32} & \mathcal{A}_{33}\end{bmatrix}}_{\mathcal{A}_{\mathrm{PPE}} \in \mathbb{C}^{3\times 3}} \begin{bmatrix}\hat{u}(y,t)\\ \hat{v}(y,t)\\ \hat{p}(y,t)\end{bmatrix}, \tag{27}$$

DOI: 10.1002/fld

where the system operators $\mathcal{A}_{11}$, $\mathcal{A}_{12}$, $\mathcal{A}_{13}$, $\mathcal{A}_{21}$, and $\mathcal{A}_{22}$ in $\mathcal{A}_{\text{PPE}}$ are the same as those in $\mathcal{A}_{\text{Pr}}$, but $\mathcal{A}_{32}$ and $\mathcal{A}_{33}$ are defined as

$$\mathcal{A}_{32} := -4\mathrm{i}\alpha y, \tag{28a}$$

$$\mathcal{A}_{33} := \frac{\partial^2}{\partial y^2} - \alpha^2. \tag{28b}$$

The wall-normal derivatives of (25) and (27) can then be approximated either using Chebyshev methods, or using finite differences on co-located or staggered meshes. In the PPE case different pressure boundary conditions can then be enforced in order to see how these differences affect the models. This will yield finite-dimensional descriptor state-space systems of the form

$$E_{\text{Pr}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = A_{\text{Pr}}\boldsymbol{\chi}(t), \tag{29}$$

and

$$E_{\text{PPE}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = A_{\text{PPE}}\boldsymbol{\chi}(t), \tag{30}$$

where $\boldsymbol{\chi} := \begin{bmatrix}\vec{\hat{u}}^* & \vec{\hat{v}}^* & \vec{\hat{p}}^*\end{bmatrix}^* \in \mathbb{C}^{n_u+n_v+n_p}$ is the state vector comprising the values of Fourier tansformed (in x) flow variables at the discretisation points. The discrete versions of $\mathcal{E}_{\text{Pr}}$ and $\mathcal{A}_{\text{Pr}}$ are $E_{\text{Pr}}, A_{\text{Pr}} \in \mathbb{C}^{(n_u+n_v+n_p)\times(n_u+n_v+n_p)}$, respectively, and $E_{\text{PPE}}, A_{\text{PPE}} \in \mathbb{C}^{(n_u+n_v+n_p)\times(n_u+n_v+n_p)}$ are the discrete versions of $\mathcal{E}_{\text{PPE}}$ and $\mathcal{A}_{\text{PPE}}$. In the co-located mesh case, $n_u = n_v = n_p$, where $n_u, n_v, n_p \in \mathbb{N}$ are the respective number of $\hat{u}$, $\hat{v}$, and $\hat{p}$ discretisation points. However in the staggered mesh case, $n_u = n_v = n_p + 1$.

4.3. 2D Channel Flow Model Formulations

In the following, a number of formulations of the linearised incompressible Navier-Stokes equations are used to deduce descriptor state-space models of 2D channel flow of the form (29) and (30). In the case of primitive variables formulations, both co-located and staggered mesh arrangements will be considered. It should be noted that due to the use of Fourier transforms in the x direction, the problem is simplified to essentially a problem in one spatial dimension, with the corresponding meshes shown in Figure 6. For the PPE formulation case a number of different pressure boundary conditions will be enforced.

Table I summarises the different models constructed. The Orr-Sommerfeld model (23) is denoted [OS], whilst [Pr1]–[Pr3] are models based on the primitive variables formulation (29), discretised in different fashions according to their identifiers in Table I.

Models [PPE1]–[PPE4] are based on the PPE formulation (30) all discretised on co-located meshes. [PPE1] implements the simple $\partial\hat{p}/\partial y = 0$ Neumann boundary condition. [PPE2] implements the divergence free velocity boundary condition in an attempt to satisfy the conditions stated by Rempfer [30]. Finally, [PPE3] (Chebyshev) and [PPE4] (finite differences) implement the Neumann pressure boundary condition discussed in both [27] and [31], with the additional Neumann condition on the normal component of the velocity in order to satisfy the divergence free condition on the boundary, as suggested in [31].

DOI: 10.1002/fld

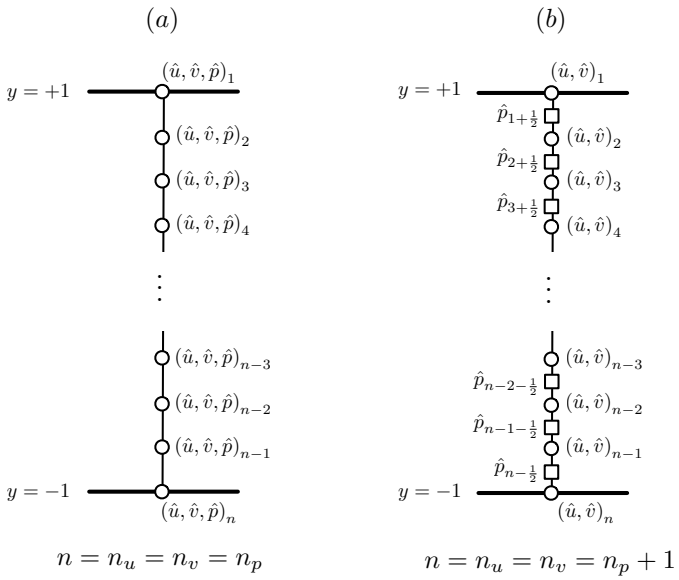

Figure 6. Channel flow computational meshes (in y direction): (a) co-located computational mesh; (b) staggered mesh.

Imposing boundary conditions in models of the form (29) and (30) is simply a matter of altering the rows of the E and A matrices corresponding to the boundary node states. In the following, the dynamics of each formulation are compared in a variety of ways. Firstly, the asymptotic behaviour of the systems is studied through comparison of the system poles and corresponding eigenfunctions. Secondly, differences in transient behaviour are explored through comparison of energy weighted pseudospectra. The various frequency responses of the systems for a particular system input and output are then computed and compared, and finally the ν-gaps between different model formulations and the benchmark Orr-Sommerfeld model are computed in order to quantify the differences in the models from a closed-loop dynamics perspective.

4.4. System Poles and Eigenfunctions

Figure 7 shows the poles of all three primitive variables models, [Pr1]–[Pr3], and both the $\hat{v}$ and $\hat{p}$ parts of the eigenfunction corresponding to the least stable mode. The poles of all three formulations lie in the same locations as those of [OS]. In all three cases the $\hat{v}$ parts of the eigenfunction agree well with that of [OS]. The $\hat{p}$ parts of the eigenfunctions of [Pr1] and [Pr3] are in agreement, however whilst the general shape of the $\hat{p}$ part of [Pr2] is similar, a non-physical sawtooth shape is clearly evident. With respect to [Pr2], the use Fourier transforms in the x direction alters the structure of the A matrix in (8) in such a way as to ensure that its transfer function exists, and so the eigenfunctions are unique (to within an arbitrary scaling). However, the use of centred finite differences in the wall-normal direction contributes to a loss of coupling between the pressure and velocity variables, hence giving rise to the non-physical pressure mode shape.

Figure 8 shows the poles of the four PPE formulations, [PPE1]–[PPE4], and both the $\hat{v}$ and $\hat{p}$ parts of the eigenfunction corresponding to the least stable mode. Whilst the poles of all four systems lie in three characteristic branches, the exact pole locations do not completely agree with those

DOI: 10.1002/fld

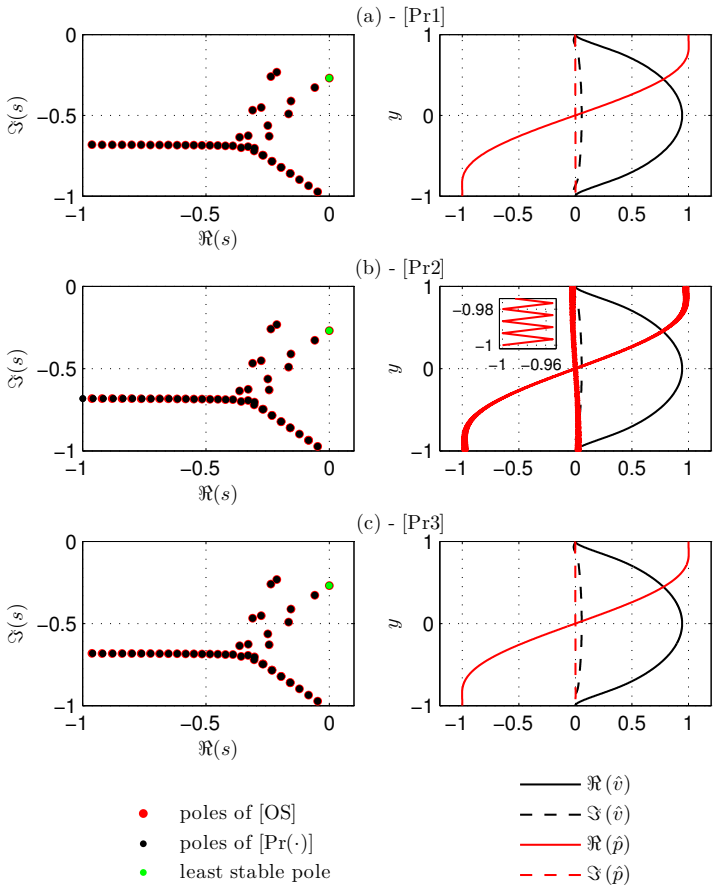

Figure 7. Poles (left) and $\hat{v}$ and $\hat{p}$ parts of eigenfunction corresponding to least stable mode (right). (a) [Pr1]; (b) [Pr2] - inset (right) is close-up of part of $\Re(\hat{p})$; (c) [Pr3].

of [OS]. The poles of [PPE1] in the S branch agree well, however in the A branch there is significant difference, not least in the failure to agree in the location of the least stable pole. There are also extra spurious poles in the P branch. The failure of this formulation to accurately match the dynamics of [OS] is unsurprising, since as was mentioned in Section 3.2, the Neumann condition (11) is unphysical and its use is primarily for the separate purpose of time-marching, as part of a predictor-corrector scheme.

In the cases of [PPE2]–[PPE4], whilst the A branches contain all the poles present in that of [OS], the branch is contaminated by additional spurious poles. Similarly to [PPE1], additional spurious poles exist in the P branch. Despite the differences in pole locations in [PPE2]–[PPE4], the $\hat{v}$ parts of the eigenfunctions agree well with that of [OS], and all three $\hat{p}$ parts match. The same cannot be said for [PPE1], however, since neither the $\hat{v}$ nor $\hat{p}$ parts agree exactly with the other models.

4.5. System Psuedospectra

Whilst the poles of a system dictate its asymptotic behaviour, the transient response is influenced by the non-orthogonality of the system eigenfunctions, as is well understood in the area of non-modal

DOI: 10.1002/fld

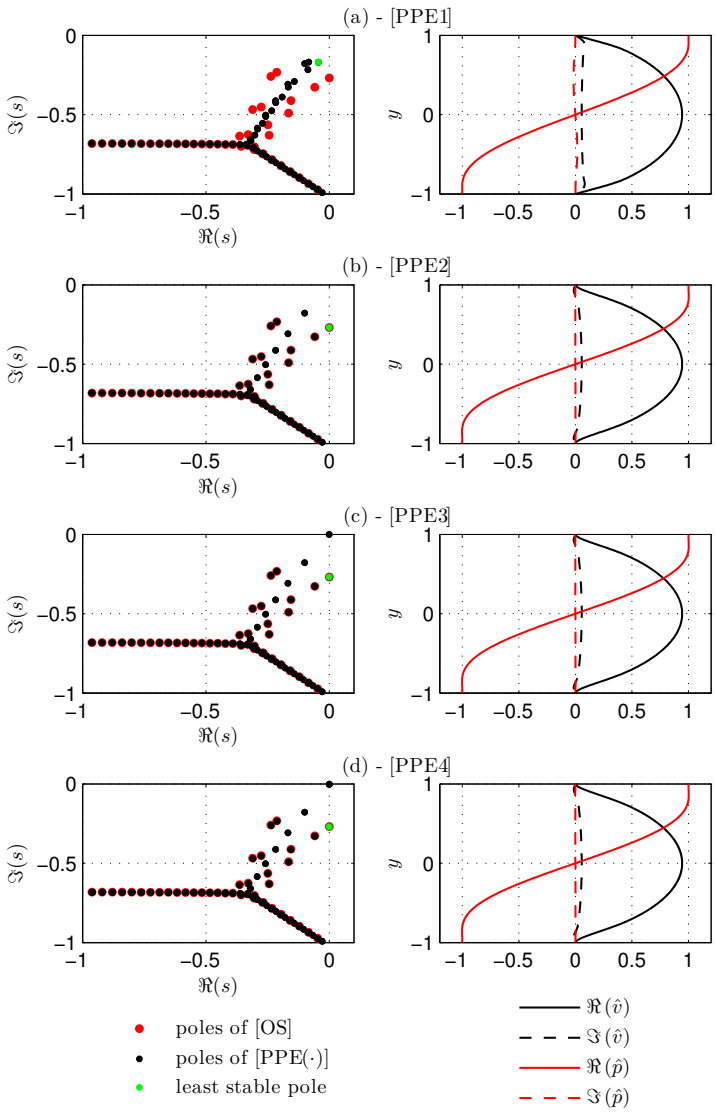

Figure 8. Poles (left) and $\hat{v}$ and $\hat{p}$ parts of eigenfunction corresponding to least stable mode (right). (a) [PPE1]; (b) [PPE2]; (c) [PPE3]; (d) [PPE4].

stability theory [45, 52, 47]. The psuedospectra Λ_ϵ, of a system provide a means of visualising this non-orthogonality in the complex plane. For a standard state-space system ($E = I$ in (3)), pseudospectra are defined as [53, 54]

$$\Lambda_\epsilon := \left\{ s \in \mathbb{C} : \left\| C(sI - A)^{-1} B \right\|_2 \geq \epsilon^{-1} \in \mathbb{R}^+ \right\}, \tag{31}$$

where C and B represent weights required to convert a measure of energy to the standard $\mathscr{L}_2$–norm [47]. This is accomplished by first defining the kinetic energy density of the

DOI: 10.1002/fld

flow perturbations $\mathscr{E}(t) \geq 0$, as in [45], as

$$\mathscr{E}(t) := \frac{1}{2\mathscr{V}} \int_{-1}^{1} \int_{0}^{\frac{2\pi}{\alpha}} u'^2(\mathbf{x},t) + v'^2(\mathbf{x},t) \mathrm{d}x\mathrm{d}y, \tag{32}$$

where $\mathscr{V} = 4\pi/\alpha$ is the integration volume. In terms of Fourier transformed velocities, this reduces to

$$\mathscr{E}(t) = \frac{1}{8} \int_{-1}^{1} \hat{u}^*(y,t)\hat{u}(y,t) + \hat{v}^*(y,t)\hat{v}(y,t) \mathrm{d}y. \tag{33}$$

This integral is evaluated numerically, and can be written

$$\mathscr{E}(t) = \begin{bmatrix} \vec{\hat{u}}^*(t) & \vec{\hat{v}}^*(t) & \vec{\hat{p}}^*(t) \end{bmatrix} \underbrace{\begin{bmatrix} \frac{1}{8}\mathscr{W}_{\hat{u}} & 0 & 0 \\ 0 & \frac{1}{8}\mathscr{W}_{\hat{v}} & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\mathscr{Q}} \begin{bmatrix} \vec{\hat{u}}(t) \\ \vec{\hat{v}}(t) \\ \vec{\hat{p}}(t) \end{bmatrix} \tag{34a}$$

$$= \boldsymbol{\chi}^*(t)\mathscr{Q}\boldsymbol{\chi}(t), \tag{34b}$$

where $\mathscr{W}_{\hat{u}} \in \mathbb{R}^{n_u \times n_u}$ and $\mathscr{W}_{\hat{v}} \in \mathbb{R}^{n_v \times n_v}$ are diagonal matrices of quadrature weights.

For the case of [OS], $\vec{\hat{u}}(t)$ does not appear in the state vector, but can be derived from the continuity equation as follows:

$$\vec{\hat{u}}(t) = \frac{\mathrm{i}}{\alpha} \mathcal{D}_y \vec{\hat{v}}(t),$$

where $\mathcal{D}_y \in \mathbb{R}^{n_y \times n_y}$ is the first order Chebyshev differentiation matrix. For [OS] (34) can then be written

$$\mathscr{E}(t) = \vec{\hat{v}}^*(t) \underbrace{\left(\frac{1}{8\alpha^2} \mathcal{D}_y^\top \mathscr{W}_{\hat{v}} \mathcal{D}_y + \frac{1}{8} \mathscr{W}_{\hat{v}} \right)}_{\mathscr{Q}_{\mathrm{OS}}} \vec{\hat{v}}(t), \tag{35}$$

where $\mathscr{Q}_{\mathrm{OS}}$ can be factored as $\mathscr{Q}_{\mathrm{OS}} = C_{\mathrm{OS}}^\top C_{\mathrm{OS}}$. Hence, the weighting matrices for computing the pseudospectra in (31) for [OS] are $B = C_{\mathrm{OS}}^{-1}$ and $C = C_{\mathrm{OS}}$.

To compute the pseudospectra of the primitive variables and PPE formulations, it is convenient to convert from descriptor to standard state-space systems. This is accomplished using the numerical algorithm outlined in [55], yielding systems of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = \check{A}_{\mathrm{Pr}}\boldsymbol{\chi}(t), \tag{36}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = \check{A}_{\mathrm{PPE}}\boldsymbol{\chi}(t). \tag{37}$$

DOI: 10.1002/fld

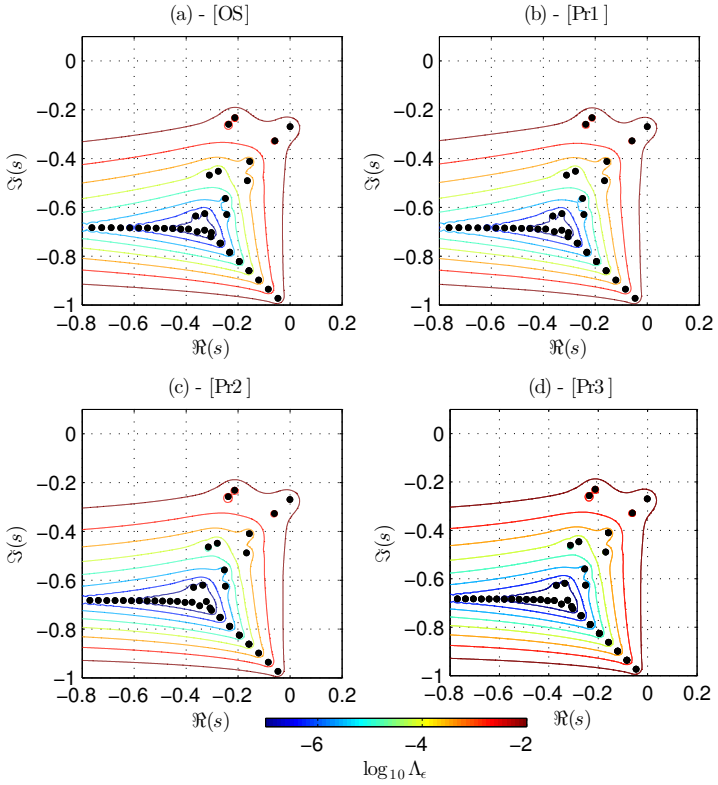

Figure 9. Energy weighted pseudospectra contour plots. (a) [OS]; (b) [Pr1]; (c) [Pr2]; (d) [Pr3].

In doing so, state transformation matrices $\mathcal{S}_{\text{Pr}}$ and $\mathcal{S}_{\text{PPE}}$ are computed, and are subsequently used to redefine $\mathcal{Q}$ in (34) for use with (36) and (37) accordingly:

$$\mathcal{Q}_{\text{Pr}} = \mathcal{S}_{\text{Pr}}^{\top}\mathcal{Q}\mathcal{S}_{\text{Pr}}, \tag{38}$$

and

$$\mathcal{Q}_{\text{PPE}} = \mathcal{S}_{\text{PPE}}^{\top}\mathcal{Q}\mathcal{S}_{\text{PPE}}. \tag{39}$$

From (38) and (39), the matrices C_{Pr} and C_{PPE} are computed and employed in a similar fashion to C_{OS} in forming appropriate energy weighting matrices.

Having constructed input and output matrices B and C, such that the models are appropriately energy weighted, pseudospectra were computed and are displayed as contour plots in Figures 9 and 10. Figure 9 shows that all three primitive variables models, [Pr1]–[Pr3], have the same pseudospectra as the Orr-Sommerfeld model [OS], implying similar transient behaviour. On the other hand, Figure 10 shows that all four PPE formulations [PPE1]–[PPE4] exhibit pseudospectra that not only differ markedly from [OS], but also differ from one another. This is understandable in the cases of [PPE1] and [PPE2], owing to their differing pole locations. However, [PPE3] and [PPE4] share the same set of poles and only differ with respect to method of spatial discretisation, and so one would expect their pseudospectra to converge upon grid refinement. This is not the case however, owing to numerical ill-conditioning of the system matrices, particularly with respect to [PPE3]. For

DOI: 10.1002/fld

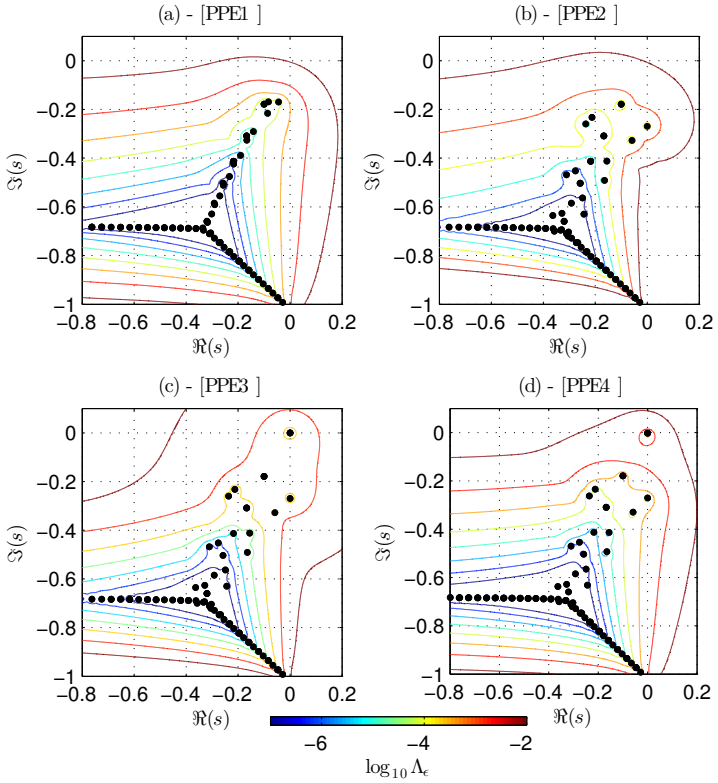

Figure 10. Energy weighted pseudospectra contour plots. (a) [PPE1]; (b) [PPE2]; (c) [PPE3]; (d) [PPE4].

example, and with respect to (31), the condition number of $\left(sI - \check{A}\right)$ for [PPE3] with $s = 0.1$ (a region where significant discrepancy exists) is $\mathcal{O}\left(10^7\right)$ greater than of that for [PPE4].

In summary, the poles, eigenfunctions, and pseudospectra reveal significant differences in the dynamics of different system formulations, pointing to significantly different open-loop behaviour. However, the question remains, what effect do these differences have when using the system formulations for feedback control design? We again stress at this point that models that are suitable for simulation are not necessarily suitable for control, and vice-versa [5, 12]. In order to gain insight into this, actuation and sensing are now defined for the 2D channel flow in order to determine the system zeros, and compare the frequency responses from actuation to sensing.

4.6. Applying Actuation and Sensing

In what follows, the choice of system actuation and measurement is somewhat arbitrary but serves to illustrate some of the subtleties of model selection for the purposes of feedback control design. As in many previous studies (see, e.g. [48, 12, 13]), we assume actuation in the form of wall transpiration, in this case at the upper wall. The actuator is modelled as a first-order system:

$$\frac{\partial}{\partial t}\hat{v}(+1,t) = -\frac{1}{\tau}\hat{v}(+1,t) + \frac{1}{\tau}q(t), \tag{40}$$

where $q(t) \in \mathbb{R}$ is the control input signal, and a time constant $\tau = 1$ is assumed for convenience. Including wall transpiration in each model formulation simply amounts to altering the rows,

DOI: 10.1002/fld

corresponding to the $\hat{v}(+1,t)$ state, of the system matrices A and E. An input matrix $B \in \mathbb{R}^{n_u+n_v+n_p}$ is also defined for each formulation. For the Orr-Sommerfeld formulation, owing to the 4^{th}-order spatial derivatives present in (18), a lifting procedure [48] is used to incorporate the effects of actuation.

The measured output $\mathbf{y}(t) \in \mathbb{C}$ is chosen to be the (Fourier transformed) pressure at the lower wall.

$$\mathbf{y}(t) = \hat{p}(-1,t). \tag{41}$$

For the primitive variables and PPE formulations discretised on a co-located mesh, the corresponding output matrices $C \in \mathbb{C}^{1\times(n_u+n_v+n_p)}$ are simply all zeros except for a one in the element corresponding to the $\hat{p}(-1,t)$ state.

For staggered mesh discretisations, since pressure nodes do not sit on the domain boundaries, linear extrapolation is used to approximate $\hat{p}(-1,t)$ based on the values of $\hat{p}(y,t)$ and $\frac{\partial}{\partial y}\hat{p}(y,t)$ at the closest node to the wall. With respect to Figure 6(b), this approximation is as follows:

$$\hat{p}(-1,t) \approx \frac{3}{2}\hat{p}_{n-\frac{1}{2}}(t) - \frac{1}{2}\hat{p}_{n-1-\frac{1}{2}}(t).$$

Therefore, the output matrices $C \in \mathbb{C}^{1\times(n_u+n_v+n_p)}$ are simply all zeros except for the elements corresponding to the $\hat{p}\left(-1+\frac{\delta}{2},t\right)$ and $\hat{p}\left(-1+\frac{3\delta}{2},t\right)$ states.

Finally, for the Orr-Sommerfeld system, pressure must be written as a function of $\hat{v}(y,t)$, since it does not appear in the state vector. Considering the Fourier transformed in x version of (16a) which reads as

$$\frac{\partial}{\partial t}\hat{u}(y,t) + \mathrm{i}\alpha\left(1-y^2\right)\hat{u}(y,t) - 2y\hat{v}(y,t) = -\mathrm{i}\alpha\hat{p}(y,t) + \frac{1}{\mathrm{Re}}\left(-\alpha^2 + \frac{\partial^2}{\partial y^2}\right)\hat{u}(y,t), \tag{42}$$

and using the fact that $\hat{u}(-1,t) = \hat{v}(-1,t) = 0$ due to the no-slip boundary condition, this reduces to

$$\hat{p}(-1,t) = \frac{1}{\mathrm{i}\alpha}\frac{1}{\mathrm{Re}}\frac{\partial^2}{\partial y^2}\hat{u}(-1,t).$$

Finally, using the continuity equation (16c), this can be written

$$\hat{p}(-1,t) = \frac{1}{\alpha^2}\frac{1}{\mathrm{Re}}\frac{\partial^3}{\partial y^3}\hat{v}(-1,t). \tag{43}$$

Hence, the output matrix $C \in \mathbb{C}^{1\times n_y}$ consists of the bottom row of $\mathcal{D}_{yyy}$, the third order Chebyshev differentiation matrix, multiplied by $1/\left(\alpha^2\mathrm{Re}\right)$.

The overall single-input single-output (SISO) systems then have the following forms

$$\frac{\mathrm{d}}{\mathrm{d}t}\vec{\hat{v}}(t) = A_{\mathrm{OS}}\vec{\hat{v}}(t) + B_{\mathrm{OS}}q(t), \tag{44a}$$

$$\mathbf{y}(t) = C_{\mathrm{OS}}\vec{\hat{v}}(t), \tag{44b}$$

DOI: 10.1002/fld

$$E_{\text{Pr}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = A_{\text{Pr}}\boldsymbol{\chi}(t) + B_{\text{Pr}}q(t), \tag{45a}$$

$$\mathbf{y}(t) = C_{\text{Pr}}\boldsymbol{\chi}(t), \tag{45b}$$

$$E_{\text{PPE}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\chi}(t) = A_{\text{PPE}}\boldsymbol{\chi}(t) + B_{\text{PPE}}q(t), \tag{46a}$$

$$\mathbf{y}(t) = C_{\text{PPE}}\boldsymbol{\chi}(t), \tag{46b}$$

with respective transfer functions

$$G_{\text{OS}}(s) = C_{\text{OS}}\left(sI - A_{\text{OS}}\right)^{-1}B_{\text{OS}}, \tag{47}$$

$$G_{\text{Pr}}(s) = C_{\text{Pr}}\left(sE_{\text{Pr}} - A_{\text{Pr}}\right)^{-1}B_{\text{Pr}}, \tag{48}$$

$$G_{\text{PPE}}(s) = C_{\text{PPE}}\left(sE_{\text{PPE}} - A_{\text{PPE}}\right)^{-1}B_{\text{PPE}}. \tag{49}$$

4.6.1. Poles and Zeros The poles and zeros of all model formulations are plotted in Figures 11 and 12. One should note the addition of a pole located at $s = -1/\tau$ in all cases owing to the presence of actuator dynamics (40). As can be seen in Figure 11, the zeros are in the same locations for the Orr-Sommerfeld model [OS], and primitive variables formulations [Pr1] and [Pr3], suggesting that these three models will exhibit the same input-output behaviour. However, whilst the majority of the zeros of [Pr2] match those of [OS], the single right half plane zero (located at approximately $s = 0.03 - 1.1\mathrm{i}$) does not match that of [OS]. This immediately suggests that their input-output behaviour differs.

As can be seen in Figure 12, the zeros of [PPE1] and [PPE2] do not agree with those of [OS]. One would therefore expect different input-output behaviour from the two systems. It should be noted that when spatially discretising the system models upon successively finer computational meshes, the matrix $(sE_{\text{PPE}} - A_{\text{PPE}})$ of [PPE2] becomes increasingly ill-conditioned, and the computed zeros fail to converge. This was not the case for any other formulation considered, suggesting that directly enforcing the divergence free velocity condition (16c) yields inherently ill-conditioned system matrices.

In the cases of [PPE3] and [PPE4], the zeros lie largely in the same locations as those of [OS], but with additional zeros lying in the same locations as the spurious poles (discussed in Section 4.4). This results in pole/zero cancellation of the spurious poles in the corresponding transfer functions and so the overall effect is that one obtains a transfer function the same as that of [OS] after computing a minimal realisation [40]. This suggests, therefore, that the two systems will exhibit the same input output behaviour.

DOI: 10.1002/fld

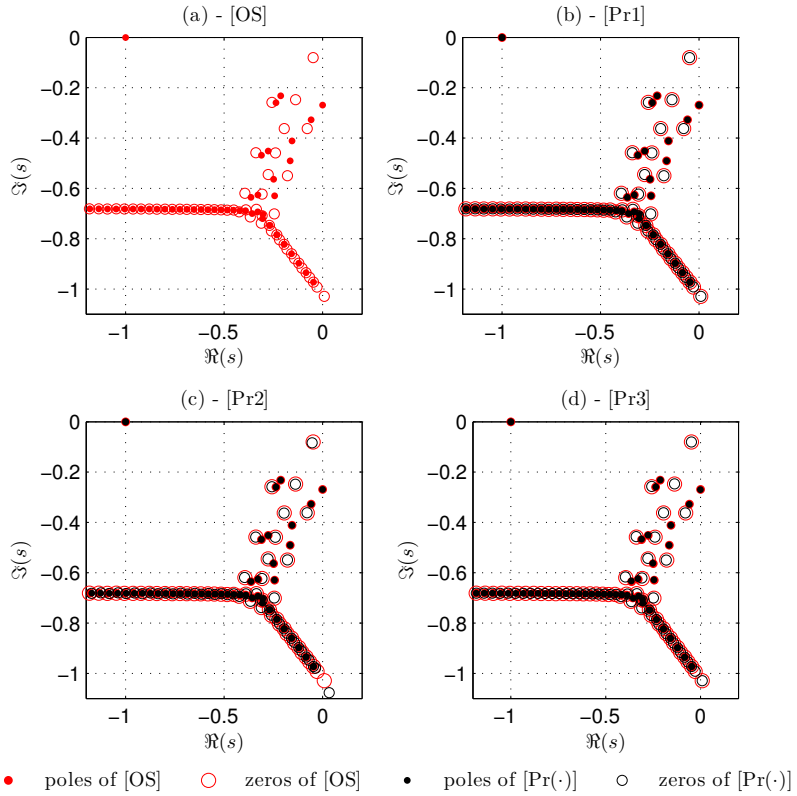

Figure 11. System poles and zeros. Poles and zeros of [OS] are shown for comparison. (a) [OS]; (b) [Pr1]; (c) [Pr2]; (d) [Pr3].

4.6.2. Frequency Response Another way of comparing input-output behaviour between systems is by comparing their frequency responses. Bode plots for each model are shown in Figures 13 and 14, where the frequency response of the Orr-Sommerfeld formulation [OS] is also plotted for comparison. As can be seen in Figure 13, the frequency responses of [Pr1] and [Pr3] match that of [OS] exactly, whilst there is an offset at low frequencies for [Pr2] due to the different location of the right half plane zero. This again exposes the problems associated with using finite difference discretisation on a co-located mesh, and also confirms that such a formulation and discretisation does not yield a model suitable for controller design as the frequency response is incorrect. The use of a staggered mesh with finite differences does indeed solve this problem. Figure 14 shows that both [PPE3] and [PPE4] have frequency response identical to that of [OS], suggesting that the PPE formulation along with pressure boundary conditions based on those discussed in [31] can produce models suitable for controller design, even when using finite difference discretisation on a co-located mesh.

The frequency response of [PPE1] shows very little agreement with that of [OS], suggesting that the PPE formulation in conjunction with the simple Neumann pressure boundary condition (11) does not yield a model that is at all representative of the actual system dynamics. Finally, whilst the frequency response of [PPE2] follows the general shape of that of [OS], it is unphysical owing to the occurrence of spurious poles and zeros. This suggests that simply enforcing the divergence free velocity condition (16c) on the boundary does not necessarily result in a suitable control model.

DOI: 10.1002/fld

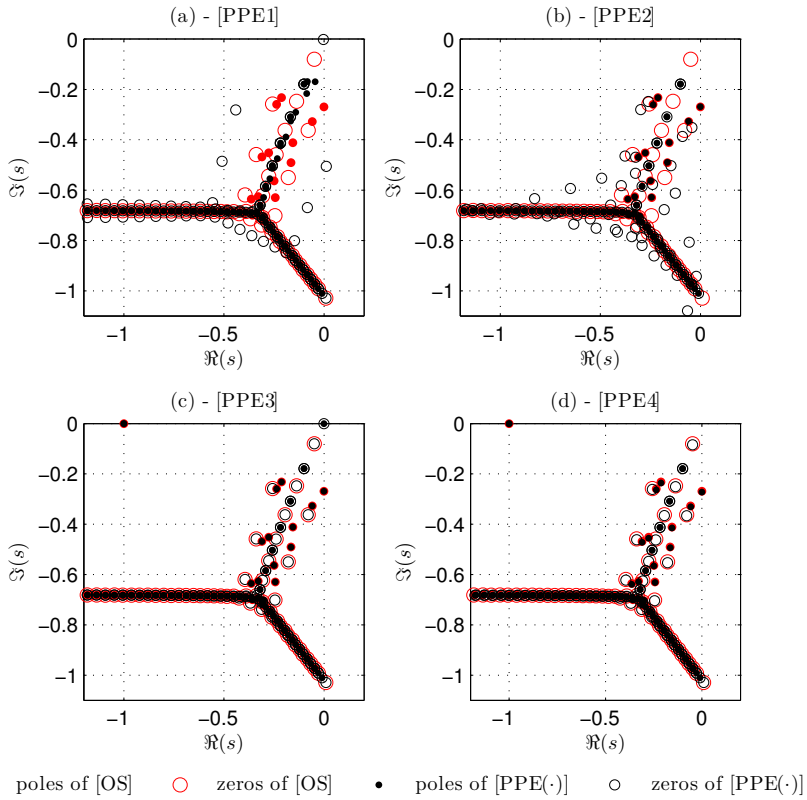

Figure 12. System poles and zeros. Poles and zeros of [OS] are shown for comparison. (a) [PPE1]; (b) [PPE2]; (c) [PPE3]; (d) [PPE4].

4.6.3. ν-gap Metric A final way of comparing the suitability of the different models for feedback control design is by considering the ν-gap metric between each of the different formulations and the benchmark Orr-Sommerfeld model. The ν-gap metric $\delta_\nu(\cdot,\cdot) : \mathscr{R}^{m\times n} \times \mathscr{R}^{m\times n} \to [0,1]$ is a measure of the 'distance' between two systems in a closed-loop sense, where a value of 0 indicates identical closed-loop behaviour, and a value of 1 indicates the opposite [56]. In terms of frequency response, the ν-gap between two systems $G_1(s)$ and $G_2(s)$ is defined as [57]:

$$\delta_\nu\left(G_1(s),G_2(s)\right) := \begin{cases} \sup\limits_{\omega\in\mathbb{R}} \bar{\sigma}\left(\Upsilon\left(G_1(s),G_2(s)\right)\right) & \text{if } \det\left(I+G_2^*(\mathrm{i}\omega)G_1(\mathrm{i}\omega)\right)\neq 0,\ \forall\omega\in\mathbb{R} \\ & \text{and wno}\det\left(I+G_2^*(s)G_1(s)\right) \\ & \qquad +\eta\left(G_1(s)\right)-\eta\left(G_2(s)\right)=0, \\ 1 & \text{otherwise,} \end{cases} \tag{50}$$

where $\Upsilon(\cdot,\cdot) : \mathscr{R}^{m\times n} \times \mathscr{R}^{m\times n} \to \mathscr{R}^{m\times n}$ is defined as:

$$\Upsilon\left(G_1(s),G_2(s)\right) = \left(I+G_2(\mathrm{i}\omega)G_2^*(\mathrm{i}\omega)\right)^{-1/2}\left(G_2(\mathrm{i}\omega)-G_1(\mathrm{i}\omega)\right)\left(I+G_1^*(\mathrm{i}\omega)G_1(\mathrm{i}\omega)\right)^{-1/2}, \tag{51}$$

the maximum singular value of $G(s)$ is denoted $\bar{\sigma}(G(s)) \in \mathbb{R}^+$, $\eta\left(G(s)\right) \in \mathbb{N}$ denotes the number of open right-half-plane poles of $G(s)$, and wno $(g(s)) \in \mathbb{N}$ denotes the winding number of the scalar transfer frunction $g(s)$ evaluated on the standard Nyquist contour.

DOI: 10.1002/fld

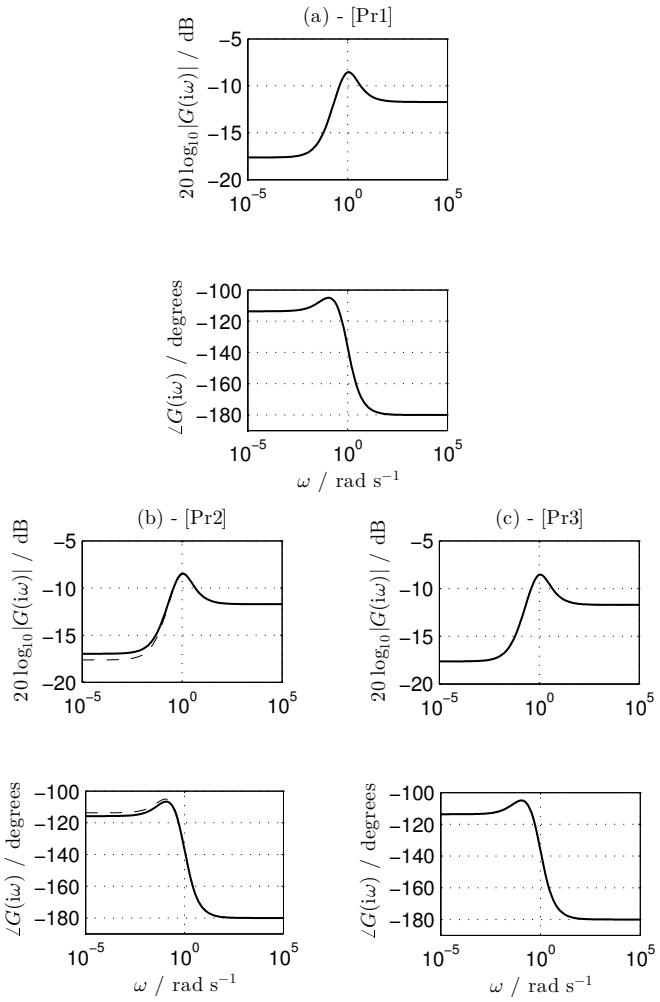

Figure 13. Bode plots (solid line). Frequency response of $G_{\mathrm{OS}}(s)$ shown for comparison (dashed line). (a) [Pr1]; (b) [Pr2]; (c) [Pr3]. Note that in (a) and (c) the dashed line is not visible as the frequency responses agree exactly.

The ν-gaps between each model formulation and the benchmark Orr-Sommerfeld model were computed, and results are presented in Table II. For models [Pr1], [Pr3], [PPE3], and [PPE4], the ν-gaps were very close to 0, and were significantly larger for [Pr2], [PPE1], and [PPE2]. It is interesting to note that whilst the frequency response of [Pr2] is broadly similar to that of the Orr-Sommerfeld model, the offset at low frequencies results in a ν-gap that is orders of magnitude greater than that of the models which agreed exactly. This is due to the differing location of the right half plane zero, and models arising from spatial discretisations that produce such spurious right half plane zeros are not suitable for feedback control design. This confirms our previous analyses.

DOI: 10.1002/fld

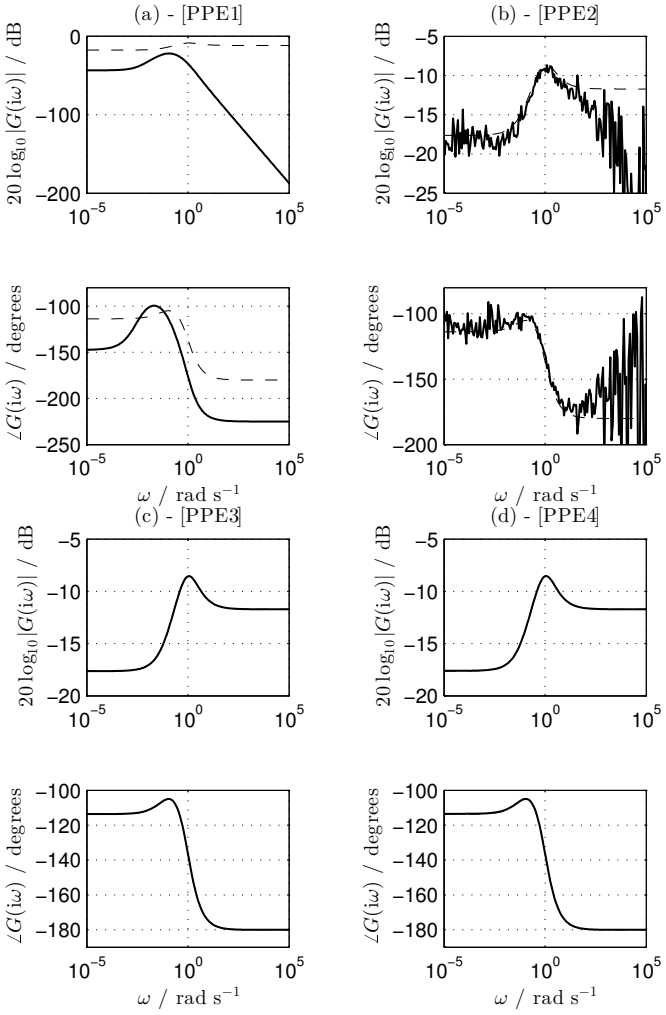

Figure 14. Bode plots (solid line). Frequency response of $G_{\mathrm{OS}}(s)$ shown for comparison (dashed line). (a) [PPE1]; (b) [PPE2]; (c) [PPE3]; (d) [PPE4]. Note that in (c) and (d) the dashed line is not visible as the frequency responses agree exactly.

5. CONCLUSIONS

In this paper, a number of different formulations of the linearised Navier-Stokes equations were compared to one another, in terms of their dynamic response, with a view towards identifying formulations that were suitable for designing feedback controllers.

Models of a 2D channel flow were constructed, and the dynamics and input-output behaviour compared to a benchmark Orr-Sommerfeld model. This showed that when using centred finite differences, the only formulation which was dynamically identical to the Orr-Sommerfeld model was the primitive variables formulation discretised on a staggered mesh. This yielded the same

DOI: 10.1002/fld

poles, zeros, psuedospectra, and frequency response, and is thus a recommended choice as the basis for designing feedback controllers, particularly for flows around complex geometries.

On the other hand, none of the models based on PPE formulations yielded the same underlying dynamics, which was clearly exposed by their differing poles and pseudospectra. However, in some cases the frequency response obtained upon defining an input and output to the system agreed with that of the Orr-Sommerfeld model, since the zeros caused pole/zero cancellation of spurious poles. This implies that some PPE-based formulations may be suitable for feedback control design. However, as the zero locations are dependent on how the inputs and outputs are defined, there is no guarantee that such pole/zero cancellation would occur for different sensing and actuation configurations, and so considerable care is advised.

A. ELEMENTS OF SINGLE NODE SUBSYSTEM DYNAMICS MATRICES

A.1. Elements of $A_{i,j}$ in (8)

$$
\begin{aligned}
a_{11} &:= -0.5\delta^{-1}\left(\bar{u}_{i+1,j} - \bar{u}_{i-1,j}\right) - 4\delta^{-2}\mathrm{Re}^{-1},\\
a_{12} &:= -0.5\delta^{-1}\left(\bar{u}_{i,j+1} - \bar{u}_{i,j-1}\right),\\
a_{21} &:= -0.5\delta^{-1}\left(\bar{v}_{i+1,j} - \bar{v}_{i-1,j}\right),\\
a_{22} &:= -0.5\delta^{-1}\left(\bar{v}_{i,j+1} - \bar{v}_{i,j-1}\right) - 4\delta^{-2}\mathrm{Re}^{-1}.
\end{aligned}
$$

A.2. Elements of $A_{i,j}$ in (9)

$$
\begin{aligned}
a_{11} &:= -0.5\delta^{-1}\left(\bar{u}_{i+3/2,j} - \bar{u}_{i-1/2,j}\right) - 4\delta^{-2}\mathrm{Re}^{-1},\\
a_{13} &:= \delta^{-1},\\
a_{22} &:= -0.5\delta^{-1}\left(\bar{v}_{i,j-1/2} - \bar{v}_{i,j+3/2}\right) - 4\delta^{-2}\mathrm{Re}^{-1},\\
a_{23} &:= -\delta^{-1},\\
a_{31} &:= 1,\\
a_{32} &:= -1.
\end{aligned}
$$

A.3. Elements of $A_{i,j}$ in (10)

$$
\begin{aligned}
a_{11} &:= -0.5\delta^{-1}\left(\bar{u}_{i+1,j} - \bar{u}_{i-1,j}\right) - 4\delta^{-2}\mathrm{Re}^{-1},\\
a_{12} &:= -0.5\delta^{-1}\left(\bar{u}_{i,j+1} - \bar{u}_{i,j-1}\right),\\
a_{21} &:= -0.5\delta^{-1}\left(\bar{v}_{i+1,j} - \bar{v}_{i-1,j}\right),\\
a_{22} &:= -0.5\delta^{-1}\left(\bar{v}_{i,j+1} - \bar{v}_{i,j-1}\right) - 4\delta^{-2}\mathrm{Re}^{-1},\\
a_{31} &:= \delta^{-1}\left(\bar{u}_{i+1,j} - 2\bar{u}_{i,j} + \bar{u}_{i-1,j}\right) + 0.25\delta^{-2}\left(\bar{v}_{i+1,j+1} - \bar{v}_{i+1,j-1} - \bar{v}_{i-1,j+1} + \bar{v}_{i-1,j-1}\right),\\
a_{32} &:= \delta^{-2}\left(\bar{v}_{i,j+1} - 2\bar{v}_{i,j} + \bar{v}_{i,j-1}\right) + 0.25\delta^{-2}\left(\bar{u}_{i+1,j+1} - \bar{u}_{i+1,j-1} - \bar{u}_{i-1,j+1} + \bar{u}_{i-1,j-1}\right),\\
a_{33} &:= -4\delta^{-2}.
\end{aligned}
$$

Prepared using fldauth.cls
DOI: 10.1002/fld

REFERENCES

1. Results of the review of the community strategy to reduce CO_2 emissions from passenger cars and light-commercial vehicles. *Technical Report*, European Commission 2007.
2. Hucho WH ((ed.)). *Aerodynamics of Road Vehicles*. 4th edn., Society of Automotive Engineers: USA, 1998.
3. Choi H, Lee J, Park H. Aerodynamics of heavy vehicles. *Annual Review of Fluid Mechanics* 2014; **46**:441–468.
4. Dahan JA, Morgans AS, Lardeau S. Feedback control for form-drag reduction on a bluff body with a blunt trailing edge. *Journal of Fluid Mechanics* 2012; **704**:360–387.
5. Aström KJ, Murray RM. *Feedback Systems*. 1st edn., Princeton University Press: Princeton, New Jersey, USA, 2008.
6. Bewley TR. Flow control: new challenges for a new renaissance. *Progress in Aerospace Sciences* 2001; **37**:21–58.
7. Kim J, Bewley TR. A linear systems approach to flow control. *Annual Review of Fluid Mechanics* 2007; **39**:383–417.
8. Sharma AS, Morrison JF, McKeon BJ, Limebeer DJN, Koberg WH, Sherwin SJ. Relaminarisation of $\mathrm{Re}_\tau = 100$ channel flow with globally stabilising linear feedback control. *Physics of Fluids* 2011; **23**.
9. Martinelli F, Quadrio M, McKernan J, Whidborne JF. Linear feedback control of transient energy growth and control performance limitations in subcritical plane Poiseuille flow. *Physics of Fluids* 2011; **23**.
10. Illingworth SJ, Morgans AS, Rowley CW. Feedback control of cavity flow oscillations using simple linear models. *Journal of Fluid Mechanics* 2012; **709**:223–248.
11. McKeon BJ, Sharma AS, Jacobi I. Experimental manipulation of wall turbulence: A systems approach. *Physics of Fluids* 2013; **25**.
12. Jones BL, Heins PH, Kerrigan EC, Morrison JF, Sharma AS. Modelling for robust feedback control of fluid flows. *Journal of Fluid Mechanics* 2015; **769**:687–722.
13. Heins PH, Jones BL, Sharma AS. Passivity-based output-feedback control of turbulent channel flow. *Automatica* 2016; **69**:348–355.
14. Aamo OM, Krstić M. *Flow Control by Feedback: Stabilization and Mixing*. 1st edn., Springer-Verlag: London, Great Britain, 2003.
15. Baramov L, Tutty OR, Rogers E. $\mathscr{H}_\infty$ control of nonperiodic two-dimensional channel flow. *IEEE Transactions on Control Systems Technology* 2004; **12**(1):111–122.
16. Barbagallo A, Dergham G, Sipp D, Schmid PJ, Robinet JC. Closed-loop control of unsteadiness over a rounded backward-facing step. *Journal of Fluid Mechanics* 2012; **703**:326–362.
17. Baramov L, Tutty OR, Rogers E. $\mathscr{H}_\infty$ control for non-periodic planar channel flows. *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, USA, 2001.
18. Foiaş C, Frazho AE. Redheffer products and the lifting of contractions on Hilbert space. *Journal of Operator Theory* 1984; **11**:193–196.
19. Jones BL. Gap metric bound construction from frequency response data. *19th World Congress for The International Federation of Automatic Control*, Cape Town, South Africa, 2014.
20. Ferziger JH, Perić M. *Computational Methods for Fluid Dynamics*. 2nd edn., Springer-Verlag: Berlin, Germany, 1997.
21. Pozrikidis C. *Introduction to Theoretical and Computational Fluid Dynamics*. 1st edn., Oxford University Press: New York, USA, 1997.
22. Chung TJ. *Computational Fluid Dynamics*. 2nd edn., Cambridge University Press: Cambridge, UK, 2010.
23. Faghri A, Zhang Y, Howell J. *Advanced Heat and Mass Transfer*. 1st edn., Global Digital Press: Missouri, USA, 2010.
24. Weickert J. Navier-Stokes equations as a differential-algebraic system 1996.
25. Brenan KE, Campbell SLV, Petzold LR. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 1996.
26. Kunkel P, Mehrmann V. *Differential-Algebraic Equations: Analysis and Numerical Solution*. European Mathematical Society: Helsinki, Finland, 2006.
27. Gresho PM, Sani RL. On pressure boundary conditions for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**:1111–1145.
28. Gresho PM. Incompressible fluid dynamics: Some fundamental formulation issues. *Annual Review of Fluid Mechanics* 1991; **23**:413–453.

DOI: 10.1002/fld

29. Sani RL, Shen J, Pironneau O, Gresho PM. Pressure boundary condition for the time-dependent incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids* 2006; **50**:673–682.
30. Rempfer D. On boundary conditions for incompressible Navier-Stokes problems. *Applied Mechanics Reviews* 2006; **59**:107–125.
31. Shirokoff D, Rosales RR. An efficient method for the incompressible Navier-Stokes equations on irregular domains with no-slip boundary conditions, high order up to the boundary. *Journal of Computational Physics* 2011; **230**:8619–8646.
32. Vreman AW. The projection method for the incompressible Navier-Stokes equations: The pressure near a no-slip wall. *Journal of Computational Physics* 2014; **263**:353–374.
33. Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids* 1965; **8**(12):2182–2189.
34. Temam R. Remark on the pressure boundary condition for the projection method. *Theoretical and Computational Fluid Dynamics* 1991; **3**:181–184.
35. Chen S, Martínez D. On boundary conditions in lattice Boltzmann methods. *Physics of Fluids* 1996; **8**:2527–2536.
36. Rani HP, Sheu TWH. Nonlinear dynamics in a backward-facing step flow. *Physics of Fluids* 2006; **18**.
37. Rani HP, Sheu TWH, Tsai ESF. Eddy structures in a transitional backward-facing step flow. *Journal of Fluid Mechanics* 2007; **588**:43–58.
38. Parkin DJ, Thompson MC, Sheridan J. Numerical analysis of bluff body wakes under periodic open-loop control. *Journal of Fluid Mechanics* 2014; **739**:94–123.
39. Dorf RC, Bishop RH. *Modern Control Systems*. 10th edn., Prentice Hall: USA, 2004.
40. Skogestad S, Postlethwaite I. *Multivariable Feedback Control*. 2nd edn., Wiley: West Sussex, UK, 2005.
41. Dai L ((ed.)). *Singular Control Systems*. Lecture Notes in Control and Information Sciences, Springer-Verlag, 1989.
42. Rempfer D. Low-dimensional modeling and numerical simulation of transition in simple shear flows. *Annual Review of Fluid Mechanics* 2003; **35**:229–265.
43. Protter MH, Weinberger HF. *Maximal Principles in Differential Equations*. Springer-Verlag: Berlin, Germany, 1984.
44. Orr WMF. The stability or instability of the steady motions of a perfect liquid and of a viscous liquid. Part I: A perfect liquid. *Proceedings of the Royal Irish Academy* 1907; **27**:9–68.
45. Butler KM, Farrell BF. Three-dimensional optimal perturbations in viscous shear flow. *Physics of Fluids A* 1992; **4**(8):1637–1650.
46. Schmid PJ, Henningson DS. *Stability and Transition in Shear Flows*. 1st edn., Springer-Verlag: New York, USA, 2001.
47. Schmid PJ. Nonmodal stability theory. *Annual Review of Fluid Mechanics* 2007; **39**:129–162.
48. McKernan J, Papadakis G, Whidborne JF. A linear state-space representation of plane Poiseuille flow for control design - a tutorial. *International Journal of Modelling, Identification and Control* 2006; **1**(4):272–280.
49. Weideman JAC, Reddy SC. A MATLAB differentiation matrix suite. *ACM Transactions on Mathematical Software* 2000; **26**(4):465–519.
50. Orszag SA. Accurate solution of the Orr-Sommerfeld stability equation. *Journal of Fluid Mechanics* 1971; **50**:689–703.
51. Mack LM. A numerical study of the termporal eigenvalue spectrum of the Blasius boundary layer. *Journal of Fluid Mechanics* 1976; **73**:497–520.
52. Trefethen LN, Trefethen AE, Reddy SC, Driscoll TA. Hydrodynamic stability without eigenvalues. *Science* 1993; **261**(5121):578–584.
53. Trefethen LN. *Spectral Methods in MATLAB*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 2000.
54. Trefethen LN, Embree M. *Spectra and Pseudospectra: The Behaviour of Nonnormal Matrices and Operators*. Princeton University Press: Princeton, New Jersey, USA, 2005.
55. Jones BL, Kerrigan EC, Morrison JF. A modeling and filtering framework for the semi-discretised Navier-Stokes equations. *Proceedings of the European Control Conference*, Budapest, Hungary, 2009.
56. Vinnicombe G. *Uncertainty and Feedback*. 1st edn., Imperial College Press: London, UK, 2000.
57. Vinnicombe G. Frequency domain uncertainty and the graph topology. *IEEE Transactions on Automatic Control* 1993; **38**(9):1371–1383.

DOI: 10.1002/fld

Table I. Summary of different 2D channel flow models.

Model identifier	Primitive variables or PPE formulation	co-located or staggered mesh	discretisation technique in y direction	$y = \pm 1$ boundary conditions
[OS]	-	co-located	Chebyshev	$\hat{v} = \frac{\partial \hat{v}}{\partial y} = 0$
[Pr1]	primitive variables	co-located	Chebyshev	$\hat{u} = \hat{v} = 0$
[Pr2]	primitive variables	co-located	finite difference	$\hat{u} = \hat{v} = 0$
[Pr3]	primitive variables	staggered	finite difference	$\hat{u} = \hat{v} = 0$
[PPE1]	PPE	co-located	Chebyshev	$\hat{u} = \hat{v} = \frac{\partial \hat{p}}{\partial y} = 0$
[PPE2]	PPE	co-located	Chebyshev	$\hat{u} = \hat{v} = \nabla \cdot \hat{\mathbf{u}} = 0$
[PPE3]	PPE	co-located	Chebyshev	$\hat{u} = \frac{\partial \hat{v}}{\partial y} = \frac{\partial \hat{p}}{\partial y} - \frac{1}{\mathrm{Re}} \frac{\partial^2 \hat{v}}{\partial y^2} = 0$
[PPE4]	PPE	co-located	finite difference	$\hat{u} = \frac{\partial \hat{v}}{\partial y} = \frac{\partial \hat{p}}{\partial y} - \frac{1}{\mathrm{Re}} \frac{\partial^2 \hat{v}}{\partial y^2} = 0$

Table II. ν-gap between different 2D channel flow model formulations and the benchmark Orr-Sommerfeld model.

[Pr1]	[Pr2]	[Pr3]	[PP1]	[PPE2]	[PPE3]	[PPE4]
0.0002	0.0112	0.0005	0.3404	0.2596	0.0002	0.0007

DOI: 10.1002/fld